Freie Universität Berlin

Volker Roth

Rechnersicherheit, SoSe 21

Übung 01

TutorIn: Oliver Wiese
Tutorium 02
Materialien: Latex, VSC, Skript

23. Juni 2023

---

1 Threat model

a)

communication history, ability to communicate

b)

1. Secret Services and Police
   Their official motivation would be to protect their country from criminals/terrorists, their real motivation might also be to protect themselves, or to advance some political agenda. Their goal is to collect all possible data on everyone, to analyze it or store it somewhere just in case it's needed later. Another goal could be to make communication between people more difficult. They have very high capabilities and resources.
2. Corporations
   They want to sell you something. They want your data to analyze it for your market preferences, or they want to sell your data further. They might be the corporation owning the communication tool. They might have high capabilities and high resources
3. Common Criminals or Hackers
   They want money. They might want your data to blackmail you, or to find sensitive information (when you are at home, what bank do you use). They might also want to blackmail the owners of the tool. They might have high capabilities, but not so high resources.

c)

d)

1. Breaking an encryption key.
   They will be able to listen in on that conversation. This is very damaging for whoever is involved in that conversation
2. DDoS Attack.
   The tool is made unavailable, nobody can communicate. This is very damaging to the image of the tools owners. Also damaging to the people who wanted to communicate.
3. Breaking the encryption algorithm.
   They can listen in on all conversation. Extremely bad.

2 Elevation of Privilege

We wrote a python script to simulate the shuffling for 4 players.

```
import random

cards =
[ [i+"_"+j for i in ['2','3','4','5','6','7','8','9','10','J','Q','K','A']]
for j in ["Spoofing","Tampering","Repudiation","InfoDisclosure","DOS"]] +
[['5_EoP','6_EoP','7_EoP','8_EoP','9_EoP','10_EoP','J_EoP','Q_EoP','K_EoP','A_EoP']]

flat_cards = [card for subl in cards for card in subl]
random.shuffle(flat_cards)
n = len(flat_cards)
p1 = flat_cards[0:n//4]
p2 = flat_cards[n//4:n//2]
p3 = flat_cards[n//2:n//2+n//4]
p4 = flat_cards[n//2+n//4:n]
print()
print(p1)
print()
print(p2)
print()
print(p3)
print()
print(p4)
```

Round 1:
Player 4: Tampering 3: Possible custom key exchange between Web Server and Auth Service.
Player 1: Tampering 6: attack the database with sql injection
Player 2: Tampering 4: low card
Player 3: Tampering 7: Bypass permissions through Auth Service, as account names are not made canonical before checking access permissions.
Points: Player 1 - 1p, Player 2 - 0p, Player 3 - 2p, Player 4 - 1p

Round 2:
Player 3: Spoofing K: There is still a default admin password in the database
Player 4: Spoofing 3: No login limit in Auth Service
Player 1: Spoofing 2: Some open ports with vulnerabilities data. E.g. SSH port.
Player 2: Spoodinf J: Credentials are stored in cookies.
Points: Player 1 - 2p, Player 2 - 1p, Player 3 - 4p, Player 4 - 2p

Round 3:
Player 3: Spoofing 10: The Webapp needs no authentication.
Player 4: Spoofing A: high card
Player 1: Spoofing Q: account recovery doesn't require disclosing the old password
Player 2: EoP Q: Webapp let clients upload pictures and customize their user profile page (like MySpace)
Points: Player 1 - 3p, Player 2 - 3p, Player 3 - 5p, Player 4 - 2p

Round 4:
Player 2: Tampering 5: Code doesn't provide timestamps or sequence numbers
Player 3: EoP K: Run commands through ssh. Ssh established in round 2 through open ports
Player 4: Tampering 10: No ACLs in Webapp
Player 1: Tampering Q: Forgoten hidden html field in the Webapp which grants a higher privilege
Points: Player 1 - 4p, Player 2 - 4p, Player 3 - 7p, Player 4 - 3p

Round 5:
Player 3: Tampering J: No ACLs in Webapp
Player 4: Spoofing 4: low card
Player 1: Tampering 4: access control decisions from Auth Service, does not use a security kernel also no ACLs
Player 2: Repudiation 2: low card
Points: Player 1 - 5p, Player 2 - 4p, Player 3 - 9p, Player 4 - 3p

Round 6:
Player 3: EoP 5: Webapp offers multiple login paths for different privileged clients
Player 4: EoP J: high card
Player 1: EoP 6: Webapp asks for unused .NET permissions
Player 2: EoP A: high card
Points: Player 1 - 6p, Player 2 - 6p, Player 3 - 10p, Player 4 - 4p

Player 3 won!

3 Docker or Virtual machine

a)

Docker is used in software development when many application and their dependencies need to be organized and isolated. Each app is put into a so called container, the process is called containerization. Each container is an instance of OS-level virtualization, the operating system gives the container its own separate user space. Docker has become extremely popular over the last few years.
Read more:
https://dzone.com/articles/docker-explained-an-introductory-guide-to-docker

b)

Virtual Box is a virtual machine. A virtual machine emulates some computer architecture(i.e x86). Basically running another computer from inside your computer. This can be used as a safe sandbox to test and isolate things without breaking anything or to just to have another operating system (f.e. running a windows VM on linux).

c)

Virtual Box emulates on the architecture level, Docker on the OS level. Containers still use the underlying OS, while a VM hosts its own OS. As such Docker can still use the underlying hardware resources directly, it is much faster and more lightweight. You can run many more instances at once. However, a virtual machine provides much more isolation and security.

d)

https://nvd.nist.gov/vuln/search

e)

There seem to be at least a 100 exploits in the above database if you search for Docker in 2020. Some of them critical:
https://nvd.nist.gov/vuln/detail/CVE-2020-29575
Also this paper from 2020 says „the number of newly introduced vulnerabilities on Docker Hub is rapidly increasing“: https://arxiv.org/abs/2006.02932
However, for Virtual Box you can also find this exploit which might allow escaping the virtual machine: https://nvd.nist.gov/vuln/detail/CVE-2020-6100 It seems much worse for Docker, but this might also be because it is used so much and so becomes a prominent target.